SPECIFICATION

Systems and Methods for Remote Management of Diagnostic Devices and Data Associated Therewith

Cross Reference to Related Applications

This application claims the benefit, pursuant to 35 U.S.C. § 119(e), of applicant's provisional U.S. Patent Application Serial No. 60/241,898, filed October 20, 2000, entitled "Service Chain Management Automation Systems and Methods," the disclosure of which is hereby incorporated by this reference for all purposes.

Background of Invention

1. Field of Invention

[0001]   The present invention relates to systems and methods for automating the tracking and monitoring of a malfunctioning item in a repair cycle. More particularly, the invention relates to systems and methods for automating the tracking and monitoring of the repair process via automated malfunction analysis, diagnostic interface standardization, centralized maintenance and distribution of test scripts used by diagnostic devices and aggregation of analysis test data.

2. Description of Related Art

[0002]

The Internet is a global network of connected computer networks. Over the last several years, the Internet has grown in significant measure. A large number of computers on the Internet provide information in various forms. Anyone with a computer connected to the Internet can potentially tap into this vast pool of

information.

[0003]   The most wide spread method of providing information over the Internet is via the World Wide Web (the Web). The Web consists of a subset of the computers connected to the Internet; the computers in this subset run Hypertext Transfer Protocol (HTTP) servers (Web servers). The information available via the Internet also encompasses information available via other types of information servers such as GOPHER, SMTP (simple mail transfer protocol), POP3 (Post Office Protocol) and FTP (file transfer protocol).

[0004]   Information on the Internet can be accessed through the use of a Uniform Resource Locator (URL). A URL uniquely specifies the location of a particular piece of information on the Internet. A URL will typically be composed of several components. The first component typically designates the protocol by with the address piece of information is accessed (e.g., HTTP, GOPHER, etc.). This first component is separated from the remainder of the URL by a colon (':'). The remainder of the URL will depend upon the protocol component. Typically, the remainder designates a computer on the Internet by name, or by IP number, as well as a more specific designation of the location of the resource on the designated computer. For instance, a typical URL for an HTTP resource might be:

[0005]   http://www.server.com/dir1/dir2/resource.htm

[0006]   where http is the protocol, www.server.com is the designated computer and /dir1/dir2/resouce.htm designates the location of the resource on the designated computer.

[0007]
Web servers host information in the form of Web pages; collectively the server and the information hosted are referred to as a Web site. A significant number of Web pages are encoded using the Hypertext Markup Language (HTML) although other encodings using the eXtensible Markup Language (XML) or the Standard Generic Markup Language (SGML) are becoming increasingly more common. The published specifications for these languages are incorporated by reference herein. Web pages in these formatting languages may include links to other Web pages on the same Web

site or another. As will be known to those skilled in the art, Web pages may be generated dynamically by a server by integrating a variety of elements into a formatted page prior to transmission to a Web client. Web servers and information servers of other types await requests for the information that they receive from Internet clients.

[0008]   Client software has evolved that allows users of computers connected to the Internet to access this information. Advanced clients such as Netscape's Navigator and Microsoft's Internet Explorer allow users to access software provided via a variety of information servers in a unified client environment. Typically, such client software is referred to as browser software.

[0009]   The Internet may serve as one exemplary communications infrastructure for use in connection with the service chain management automation according to the present invention, as described in greater detail below. The entire service chain encompasses the entire process from an end user identifying a problem with a device to the return of the repaired device, or a replacement device where repair is either not feasible or cost effective.

[0010]   One aspect of the overall service chain is the diagnosis of the problem with the device. For many electronic devices or other types of items, such as mobile phone handsets, diagnostic instruments, or analyzers, are connected to the device and perform one or more tests. Technicians use analyzers to run a series of tests against a phone that is in for repair. Theses tests are diagnostic in nature and will look at various system functions within the phone to enable the technician to determine where problems exist with the phone.

[0011]   A series of one or more individual tests is called a script. A manufacturer, carrier, Authorized Service Center (ASC) or re-manufacturer may build a series of scripts for each phone type. A technician will determine what script to use to test the phone depending on the problem reported with the phone. A technician can run multiple scripts against a phone when looking for problems, or they may run one script and then several individual tests to get clarification of the problem.

[0012]   A variety of analyzers such as those manufactured by companies such as Agilent, Anritsu, and Acterna are used in the mobile phone industry to test handsets that are returned for repair. The analyzers are used to discover problems with the electronic and software components of a handset. The data readouts from each test on a handset are of huge value to the repair technician to make an immediate diagnosis of non-visible problems with the phone. However, as there is no existing method to trap and warehouse this data, aggregation of the data is not typically done.

[0013]   An additional problem faced by users of the analyzers is getting timely updates to the test scripts that are used to test each phone. A test script may specify one or more tests to be performed by the analyzer on a malfunctioning device. The required tests and the pass/fail limits of the tests are set by the manufacturers, who then rely on the analyzer manufacturers to distribute updated software containing the new scripts to all users. This is at best done haphazardly. A centralized location for updated scripts, or a better method of gaining access to updated scripts is highly desired.

[0014]   The systems and methods according to the present invention address these and other issues related to diagnostic procedures in present service chains.

Summary of Invention

[0015]   The present invention relates to systems and methods for automating the tracking and monitoring of the repair process via automated malfunction analysis, diagnostic interface standardization, centralized maintenance and distribution of test scripts used by diagnostic devices and aggregation of analysis test data. A typical system according to the present invention may include a system processor and a system data store. In certain embodiments, the system processor may be used to implement methods according to the present invention. Further, computer-readable media, such as magnetic and optical disks, primary storage devices such as RAM and ROM, etc., may be used to store instruction that cause a computer to execute methods according to the present invention.

[0016]
According to the present invention, diagnostic data is received from one or more diagnostic devices and, in some embodiments, may be stored in a data store.

Diagnostic data may be sent in response to polling by a centralized system processor or as initiated by individual diagnostic devices. Reports analyzing a subset of the diagnostic data received are generated and outputted. Typically, reports will be based upon item type, malfunction type, diagnostic device type, item manufacture and/or test/tests performed. Reports may be generated either in response to a report request or other trigger event or at periodic intervals.

[0017]   In some embodiments, one or more system diagnostic devices may be included in the system in addition to, or instead of, diagnostic devices communicating with an aggregation and reporting environment according to the present invention. Typically, such system diagnostic devices will include a testing element or device responsible for performing tests on malfunctioning items, a diagnostic processor that communicates diagnostic data from the testing device to the system processor, and, in some cases, a diagnostic data store that temporarily or permanently stores data generated by the testing device.

[0018]   Further, in addition to diagnostic data, scripts containing instructions for use by diagnostic devices for running tests may be accessed and/or generated via a central location in some embodiments. In certain system embodiments, the scripts may be stored either a separate script data store and/or as a portion of a system data store. Generation of new scripts may be accomplished in some particular embodiments; such new scripts may be generated based upon existing scripts, typically supplied, at least initially, by diagnostic device manufacturers, and upon diagnostic data accumulated from the diagnostic devices. Scripts, either newly generated or pre-existing, may be distributed to diagnostic devices via the present invention.

[0019]

In another aspect of the present invention, centralized management of diagnostic devices may also include development and distribution of standardized interface for driving diagnostic devices by diagnostic technicians. Such interfaces would be independent of specific diagnostic device manufacturer but might depend upon the type of item being tested and/or the symptoms exhibited by the item being tested. In one embodiment, standardization may be accomplished through instruction contained within scripts distributed via the present invention. Scripts could be interpreted by

software local to a specific diagnostic device which would generate an interface for driving the specific diagnostic device. Benefits derived from such a standardization of interface include commonality of test names across all item models within a given item type, ensuring that only the most up-to-date scripts are being used on analyzers by removing the issues around manually updating analyzer software and script data and giving consistent interfaces to users independent of the analyzer is being used and, in some embodiments, independent of the model of the item being tested.

[0020]   The data aggregated according to the present invention is valuable to all parties involved in the repair of the malfunctioning item. In the instance where the item subject to repair is a mobile phone, the data will be useful for carriers, authorized service centers (ASCs) and manufacturers. For carriers, the aggregated failure data may be used during negotiations with manufacturers, as support for warranty claims and to more accurately project future work flows. For ASCs, the aggregated failure data may be used as support for warranty claims and to more accurately project future work flow. For manufactures, the aggregated data may be used to improve customer service by providing faster payout of warranty claims, to improve warranty claim submission accuracy from customers and suppliers, to obtain faster, more accurate failure data on particular item models, and to save money by not having to re-do tests already run by carriers/service centers. The ASC and manufacturer benefits may be realized for item types other than mobile phones.

[0021]   Additional advantages of the invention will be set forth in part in the description which follows, and in part will be obvious from the description, or may be learned by practice of the invention. The advantages of the invention will be realized and attained by means of the elements and combinations particularly pointed out in the appended claims. It is to be understood that both the foregoing general description and the following detailed description are exemplary and explanatory only and are not restrictive of the invention, as claimed.

Brief Description of Drawings

[0022]
     The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate one embodiment of the invention and together with the

description, serve to explain the principles of the invention.

[0023]   FIG. 1 is a diagram representing a system according to the present invention wherein diagnostic devices communicate with an aggregation and reporting environment via the Internet.

[0024]   FIG. 2 is a diagram representing a system according to the present invention wherein diagnostic devices are a part of the local communication network of the aggregation and reporting environment.

[0025]   FIG. 3 is a diagram representing a system according to the present invention wherein diagnostic devices are both within and without the local communication network of the aggregation and reporting environment.

[0026]   FIG. 4 is a flow chart of the process of transferring test data from a diagnostic device to an aggregation server in a typical push embodiment.

[0027]   FIGs. 5-8 depict example reports that might be generated by an environment according to the present invention.

Detailed Description

[0028]   Preferred embodiments of the invention are now described in detail. Referring to the drawings, like numbers indicate like parts throughout the views. As used in the description herein and throughout the claims that follow, the meaning of "a," "an," and "the" includes plural reference unless the context clearly dictates otherwise. Also, as used in the description herein and throughout the claims that follow, the meaning of "in" includes "in" and "on" unless the context clearly dictates otherwise.

[0029]   Ranges may be expressed herein as from "about" one particular value, and/or to "about" another particular value. When such a range is expressed, another embodiment includes from the one particular value and/or to the other particular value. Similarly, when values are expressed as approximations, by use of the antecedent "about," it will be understood that the particular value forms another embodiment. It will be further understood that the endpoints of each of the ranges are significant both in relation to the other endpoint, and independently of the other

endpoint.

[0030]   The description below describes analyzers used in the mobile phone industry; however, this specific item type is to be taken as exemplary. It will be apparent to those of skill in the art that the principles involved with analyzers for a particular item type will apply equally to analyzers for other types of devices including, without limitation, computers; hand-held electronic devices such as personal data assistant devices, pagers, calculators, clocks, radios, etc.; automobile components; medical devices; networking communications equipment such as switches, routers, modems, broadband communication enabling devices, etc.; standard telephones; televisions; or other similar devices, particularly computer controlled devices.

[0031]   Variations on typical environments according to the present invention are seen in FIGs. 1-3. FIG. 1 diagrammatically represents an aggregation and reporting environment 100 servicing as a diagnostic device data aggregation and management center where diagnostic devices 120 do not participate in the local communication network 140 of the environment 100. This arrangement would be typical of an embodiment of the present invention wherein aggregation, reporting and management services were provided on an Application Service Provider (ASP) basis. One or more diagnostic device owners would outsource the management of owned diagnostic devices 120 to the ASP providing services via environment 100. Alternative, FIG. 1 could also be used by a large organization having diagnostic devices located at geographically and/or logically diverse locations. The variation of FIG. 2 provides an environment 200 similar to that of FIG. 1; however, diagnostic devices 210 form a part of the environment 200. As depicted, such diagnostic devices 210 could participate in the same communications network 140 as the remainder of the environment. Finally, FIG. 3 depicts a combined variation of FIGs. 1 and 2 where diagnostic devices 120, 210 appear both within and without the environment 200.

[0032]
          Referring to FIGs. 1-3, diagnostic device management and data aggregation and reporting environments 100, 200 may include a server cluster 150 of one or more servers (e.g. 152, 154, 156) that provides management, aggregation and reporting functionality. These or other servers may support access by users requesting and/or

reviewing reports 110 and by diagnostic devices 120, 210. Access to the environment by these various users/devices may be via any suitable communication channel, which in a typical embodiment will be a computer network such as the Internet 130 and/or Ethernet 140. In other environments, access may be via other forms of computer network, direct dial-up connection, dedicated connection, or other suitable channel as would be known to those skilled in the art. In some embodiments the access channel may provide security features; for instance, a secure socket layer (SSL) may be used with respect to an embodiment using the Internet 130 as the access communication channel. The one or more servers may include or connect to a data store 180 for storing aggregated data, and in some embodiments, scripts for the environment.

[0033]   The various components of the environment 100, 200 may communicate with each other through any suitable communication architecture including, but not limited to, a computer network such as a Ethernet 140, token ring network or the Internet 130; a direct connection such as a bus connection, parallel or serial connection, null modem connection or wireless connection utilizing an appropriate communication protocol such as BLUETOOTH; and a dial-up connection. In embodiments where a single computer may provide all functional components, the communication may occur via bus connections, inter-process communication, shared files or some combination of these methods or other commonly utilized communication mechanisms.

[0034]   The architecture seen in FIGs. 1-3 use the Internet 130 and an Ethernet 140 as communication channels allowing access to the environment by various users 110 and diagnostic devices 120, 210. The environments uses a computer network such as the depicted Ethernet 140 to allow communication among the components of the environment; a router 135 is included in the environment to manage such communication within the internal network as well as managing the interface between the internal network and the Internet 130. The functionality of the environment is spread among a server cluster 150, a data store 180 and, in some embodiments, a load balancing device 145. Where a load balancing device 145 is present, the device may be responsible for allocating and managing distribution of access among various elements within the server cluster 150 and/or the data store 180. Users may access

the environment through standard Web browser software or via specialized access software adapted for interfacing with the aggregation and reporting environment 100, 200.

[0035] In some embodiments, diagnostic devices 120 simply communicate with the environment 100. Whereas in others, diagnostic devices 210 form a part of the environment 200; in some such embodiments, additional diagnostic devices 120 may also communicate with the environment 210. In FIGs. 1-3, the communication by diagnostic devices 210 within the environment is shown as via Ethernet while communication by diagnostic devices 120 outside the environment 100, 200 is shown as via the Internet. It will be understood that the invention is not limited to this mode of communication; rather, diagnostic devices 120, 210 whether within or without the environment 100, 200 may utilize any suitable communication channel including those describe above with respect to intercomponent communication within the environment 100, 200. The diagnostic devices may be in constant or intermittent communication with one or more servers in the environment. Communication may be initiated by any of the diagnostic devices depending upon circumstances or by other elements within the environment, particularly servers within the server cluster.

[0036] The server cluster 150 provides the aggregation and reporting functionality of the environment 100, 200. In one embodiment, the server cluster 150 may be divided into access servers and application servers where the access servers provide electronic access functionality such as by electronic mail server(s) and/or Web server(s) and the application servers provide the aggregation, analysis and reporting functionality. In one embodiment, the one or more servers (e.g. 152, 154, 156) in the server cluster 150 may be supported via Intel-compatible hardware platforms preferably using at least a PENTIUM III class microprocessor (Intel Corp., Santa Clara, CA). The hardware platform would have an appropriate operating system such as WINDOWS 2000 Server (Microsoft, Redmond, WA), WINDOWS/NT Server (Microsoft, Redmond, WA), Solaris (Sun Microsystems, Palo Alto, CA), or LINUX (or other UNIX variant). Depending upon the hardware/operating system platform, appropriate server software may be included to support the desired application, email and Web server functionality. In one embodiment, the Web server functionality may be provided via an Internet Information

Server (Microsoft, Redmond, WA). The email services may be supported via an Exchange Server (Microsoft, Redmond, WA). Application servers in some embodiments may be iPlanet Application Servers (iPlanet E-Commerce Solutions – A Sun – Netscape Alliance, Mountain View, CA), WebSphere Servers (International Business Machines, Armonk, NY) or Citrix MetaFrame (Citrix Systems, Inc., Ft. Lauderdale, FL). In one embodiment, the business application services may be provided through programmed pages on the Web server; such pages may use ActiveX, VBScript, Java Applet and/or Servelet technology to provide server side business logic and may use ActiveX or JavaScript to support client side business logic.

[0037]   The data store 180 provides for the storage and, potentially, the management of the data required by the environment. A typical data store will include one or more storage devices (e.g. 182, 184, 184, 186), and in some embodiments, may include one or more data servers (e.g. 160). Information concerning different users, information related to diagnostic devices and aggregated diagnostic data are stored in the data store 180. The architecture of the data store 180 may vary significantly in different embodiments. In several embodiments, database(s) are used to store and manipulate the data; in some such embodiment, one or more relational database management systems, such as SQL Server (Microsoft, Redmond, WA), ACCESS (Microsoft, Redmond, WA), ORACLE 8i (Oracle Corp., Redwood Shores, CA), Ingres (Computer Associates, Islandia, NY), or Adaptive Server Enterprise (Sybase Inc., Emeryville, CA), in connection with a variety of storage devices/file servers. In other embodiments, the data store 180 may use database systems with other architectures such as object-oriented, spatial, object-relational or hierarchical or may use other storage implementations such as hash tables or flat files or combinations of such architectures.

[0038]   Diagnostic devices 120, 210 whether part of the aggregation environment or simply communication with it will include a testing component used to perform tests on a malfunctioning item. In some embodiments, diagnostic devices 120, 210 may include a diagnostic processor and/or a diagnostic data store. A single diagnostic processor and/or a single diagnostic data store may be shared across multiple diagnostic devices in some instances. Diagnostic device may also in certain instance

include or have associated with them a diagnostic display device for displaying an interface for driving the diagnostic device, for displaying test results and/or for displaying reports on aggregated data generated according to the present invention. The testing component/device typically is responsible for running one or more tests on an item of a specific test item type. The resultant test data may be stored temporarily or permanently in a diagnostic data store if present; the test device or the diagnostic processor may be responsible for placing the test data in the appropriate diagnostic data store.

[0039]   Aggregation of data occurs through the transfer of data from diagnostic devices 120, 210 to a processor within a server (e.g. 152) of the server cluster 150 inside aggregation and reporting environment 100, 200. In different embodiments, the transfer of data may be in initiated exclusively from pulling by a server 150, exclusively from pushing from the diagnostic devices 120, 210, or a combination of pulling and pushing.

[0040]   In push based embodiments, diagnostic devices 120, 210 initiate aggregation of data by sending data to a server in the server cluster 150. If the embodiment is purely push based, transmission of data is independent of prompting from the receiving server. Diagnostic devices 120, 210 may typically include a diagnostic processor responsible for transmitting the data as it is developed, upon demand or at scheduled periodic intervals. The transfer may be accomplished via a server running on such a diagnostic processor, such as an email server set to send data files using SMTP protocol. In another environment, a specialized client may be used to transfer the data via a server running on one or more of the server systems in the server cluster 150. Typical examples would be a Web server and/or an FTP server where the specialized client running on the diagnostic processor would transfer data files using a POST method transfer (HTTP or HTTPS) or a PUT command, respectively. In some embodiments, raw data from the analyzer may be sent. In other embodiment, a diagnostic processor may receive the raw data and convert it into a more appropriate format such as HTML, XML, XSL, SGML, some combination thereof, or other suitable format. In some embodiments, the data may be stored locally in a diagnostic data store in either raw or formatted form. Such storage may be for temporary intervals

such as until transferred to the aggregation environment or a fixed time period or on a more permanent basis.

[0041]   In pull based embodiments, a server (e.g. 152) within the server cluster 150 of the aggregation and reporting environment 100, 200 pulls data from the diagnostic devices 120, 210. If the embodiment is purely pull based, transmission of data is independent of prompting from the diagnostic devices 120, 210. Some pull based embodiments may include a diagnostic processor controlling and supporting one or more diagnostic devices 120, 210. The diagnostic processors may support a server for receiving requests from a server in the server cluster 150. Such servers may be Web servers, FTP servers, GOPHER servers, WAIS servers, email-based file retrieval server, or other suitable server software for transmitting data files in response to a request. In other embodiments, diagnostic devices 120, 210 may store data to a diagnostic data store directly accessible by the aggregation environment 100, 200. A system processor within a server of the server cluster 150 may pull test data in various ways. In some embodiments, polling of diagnostic devices may occur at periodic intervals. In addition, or alternatively, polling may result from the receipt of a request for a report aggregating test data. In one particular embodiment, diagnostic devices 120, 210 may store data locally in a diagnostic data store accessible to their own Web servers and exposed programmatically (rather than visually) for periodic collection using a Web Service approach such as Microsoft's forthcoming .NET technology. In some embodiments, raw data from the diagnostic device 120, 210 may be pulled. In other embodiment, a diagnostic processor may receive the raw data and convert it into a more appropriate format such as HTML, XML, SGML or other suitable format. In some embodiments, the data may be stored locally in a diagnostic data store in either raw or formatted form. Such storage may be for temporary intervals such as until pulled by the aggregation environment or a fixed time period or on a more permanent basis.

[0042]   In a particular push embodiment, a software product is associated with each diagnostic processor associated with one or more diagnostic devices 120, 210. The software product may reside in any computer readable storage media. While in use this software will reside in a computer readable storage media accessible to each respective diagnostic processor such as a diagnostic store that may include one or

more forms of primary storage such as cache memory, Random Access Memory (RAM) or Read Only Memory (ROM) and/or one or more forms of secondary storage such as removable or fixed, optical or magnetic disk, paper or magnetic tape or punch card.

[0043]   FIG. 4 depicts the process performed by this software product. In step 410, the software product monitors for test data. A determination is made whether new test data is available or not in step 415. If test data is not available, the software continues to monitor for test data at step 410. If test data is available, a connection is established to a server in the server cluster 150 in step 420. After the connection is established, the software transmits the data to the server via the established connection in step 430.

[0044]   In step 410, the software may monitor for test data in several ways. In one embodiment, the software monitors one or more locations of one or more diagnostic data store for the appearance of test data. If the diagnostic processor is associated with multiple diagnostic devices, the software product may monitor multiple locations within the diagnostic data store or multiple diagnostic data stores; alternatively, multiple instantiations of the software product may run on a single multitasking diagnostic processor where each instantiation monitors one or more locations in one or more diagnostic data stores. Test data may be placed in an appropriate location in a diagnostic data store either directly by the testing component of a particular devices or indirectly where the testing component communicates the test data to a diagnostic processor which places the received test data in the diagnostic data store. In other embodiments, the software may monitor the testing element directly for generation of test data. Again, if multiple diagnostic devices are associated with the diagnostic processor, the software may monitor multiple associated diagnostic devices and/or run multiple instantiations of the software where each instantiation monitors one or more of the multiple diagnostic devices.

[0045]
     In step 415, a determination is made as to whether new test data is available for transmission to the server. The check may occur at periodic intervals or upon demand from another source such as the testing component of a diagnostic device. The check may be, in one embodiment, a check on a directory to determine if a data file has

been placed in it. In another embodiment, the determination may be made by examining a communication channel associated with the one or more testing elements.

[0046]   If test data is available to be pushed, a connection is established with a server in the server cluster 150 in step 420. Typically, the connection will be over a computer network such as Ethernet 140 or the Internet 130; however, in other embodiments, the connection could be established via another form of computer network such as a token ring network; a direct connection such as a bus connection, parallel or serial connection, null modem connection or wireless connection utilizing an appropriate communication protocol such as BLUETOOTH; and a dial-up connection. In embodiments where a single computer may provide all functional components of both the server cluster 150 and the diagnostic processor, the communication may occur via bus connections, inter-process communication, shared files or some combination of these methods or other commonly utilized communication mechanisms.

[0047]   Once the communication channel is established, the data is transferred in step 430. In one embodiment, the software product acts as a specialized Web client communicating with the server via the HTTP protocol. In such an embodiment, the client may transfer the data using a POST method request of the test data. In some circumstances, the test data, or portions thereof, could be transferred as parameters of a GET method request; however, this approach is more limited than the POST method. As an alternative, the software product may act as an FTP client and transfer the test data via a PUT command to an FTP server running in the server cluster 150.

[0048]   Data in the environment may be accessed via reports. Typically, reports will be generated by a server (e.g. 154) from the server cluster 150. Report generation may occur as a result of a variety of circumstances. Reports may be generated upon demand such as via a request by a report viewer 110 or an automated system such as a warranty claim submission and tracking subsystem or a script generation system, both described in greater detail below. In addition, reports may be generated on a periodic basis.

[0049]   Reports may be generated based upon a variety of criteria associated with the

data. The reports could use criteria such as item type being tested, malfunction type, diagnostic device type, item manufacturer and specific test performed. FIGs. 5-8 present some reports that could be generated in typical embodiments. These sample reports will typically cover a specific range of time, which, in some embodiments, may be selected or specified by a report requester.

[0050]   FIG. 5 depicts a sample report of analyzer usage showing diagnostic devices by location and reporting number of tests performed by individual devices. FIG. 6 is a sample report providing a report on malfunctions broken down by item type model. in this sample, the item type is mobile phones, the report is further organized by item manufacturer as well as specific model. FIG. 7 shows a sample report displaying data on results of particular tests organized by diagnostic device. The final example, FIG. 8, displays a report aggregating failures by item manufacturer, where listings for each manufacturer are organized by test performed.

[0051]   In some embodiments, report viewers 110 can contact an access server in the server cluster 150 to request a report. The access server in question will typically be a Web server; however, in some embodiments, the access server may be an email server, WAIS server, GOPHER server or FTP server. Generated reports may be outputted to requesting report viewers via the appropriate access server in the server cluster 150. The request for the report from a report viewer 110 may, in certain embodiments, be the event that causes the report generation. In other embodiments, requests simply access previously generated reports without necessarily serving as the impetus for generation. Report viewers 110 need not be the only type of report recipient. Reports may, in some embodiments, be requested from and/or received by a diagnostic device selected from those communicating with the server. Reports may also be requested and/or received by other automated systems that further process and/or analyze the generated report.

[0052]
          Outputted reports may be formatted in an appropriate output format such as an appropriate formatting language (e.g. HTML, XML, SGML, XSL, combinations thereof, etc.) or an industry standard format (e.g. spreadsheet (Excel, Lotus, etc.), word processing (Word, WordPerfect, etc.), database (e.g. DB2, Access, SQL, etc.) or

electronic document (e.g. PDF). The output report will be received by the recipient and displayed using a user display device such as a printer, facsimile, end user computer or a display device associated with one of the diagnostic devices 120, 210, processed further or both.

[0053]   Reports may also be outputted directly to automated processing systems such as a warranty claim submission, analysis and/or tracking system or subsystem, insurance claim processing system and/or a script generation system. In these cases, the generated report is received by the particular automated system and processed further. Reports may be outputted for further processing in commercially available spread sheet, word processing and/or database applications.

[0054]   Finally, reports may be distributed in an automated fashion. Specific recipients may be associated with particular report types. Upon generation of a report of the particular report type, a copy is distributed to all designated recipients. Report delivery may be accomplished via any suitable delivery platform including without limitation electronic mail, facsimile, remote printing, etc.

[0055]   In some embodiment of the present invention, centralized management of diagnostic devices may also include development and distribution of standardized interface for driving diagnostic devices by diagnostic technicians. Such interfaces would be independent of specific diagnostic device manufacturer but might depend upon the type of item being tested and/or the symptoms exhibited by the item being tested. In one embodiment, standardization may be accomplished through instruction contained within scripts distributed via the present invention.

[0056]   In script managing embodiments, a script data store will typically be present at the centralized authority for storing scripts for dissemination or retrieval. Such embodiments may have a similar form to those depicted in FIGs. 1-3. In such embodiments, the script data store may be a part of the system data store 180 or may be separate. A separate script processor, communicating with the script data store and/or the system data store as well as the diagnostic devices 120, 210 that are already in communication with a system processor of the server cluster 150, may also be part of a typical system; alternatively, a system processor in a server (e.g. 156) of

the server cluster 150 may provide this functionality in certain embodiments.

[0057]   Scripts could be interpreted by software local to a specific diagnostic processor associated with one or more diagnostic devices, which would generate an interface for driving the specific diagnostic device. Typically, the interface would be presented via a diagnostic display in communication with the diagnostic processor. In some instances, the diagnostic processor and display could be components of a computer system used to control one or more diagnostic devices. In other embodiments, the diagnostic processor, the diagnostic display or both could be physically integrated into a specific diagnostic device. Benefits derived from such a standardization of interface include commonality of test names across all item models within a given item type, ensuring that only the most up-to-date scripts are being used on analyzers by removing the issues around manually updating analyzer software and script data and giving consistent interfaces to users independent of the analyzer is being used and, in some embodiments, independent of the model of the item being tested.

[0058]   Script management within such embodiments may occur in push, pull or combined push-pull embodiments. Push embodiments involve dissemination of scripts from a centralized script authority independent of individual diagnostic devices. Pull embodiments involve individual diagnostic devices retrieving scripts from a centralized script authority independent of events at the script authority. Combined embodiments may use aspect of both push and pull embodiments.

[0059]

In push based embodiments, centralized script authority initiate script dissemination by sending data to one or more diagnostic devices. In embodiments using an architecture as depicted in FIGs. 1-3, the centralized script authority functionality may be provided by a server (e.g. 156) of the server cluster 150 in connection with the system data store 180. In other embodiments, the processing and/or data storage functionality, however, may be accomplished via other components not depicted. If the embodiment is purely push based, transmission of data is independent of prompting from the receiving diagnostic devices. Diagnostic devices may typically include a diagnostic processor responsible for receiving the script data. The centralized script authority may transmit scripts to diagnostic devices

either periodically or as the result of some server based event such as an update to the scripts in the script data store. The transfer may be accomplished via a server running on a system associated with the centralized script authority, such as an email server set to send data files using SMTP protocol. In another environment, a specialized client may be used to transfer the data via a server running on one or more of the diagnostic devices. Typical examples would be a Web server and/or an FTP server where the specialized client running at the centralized authority would transfer data files using a POST method transfer (HTTP or HTTPS) or a PUT command, respectively. In addition, the centralized script authority may provide multiple methods for dissemination of scripts. In such embodiments, each diagnostic device may designate one or more methods for receiving the script data. These designated methods may have preference levels defining the most/least preferred method of script delivery. In a typical push base embodiment, a data store at the centralized authority may be used to store information associated with each diagnostic device. Such information may include script delivery method preference, device manufacturer and item type for which the device may be used. The centralized authority may, in a typical embodiment, use this information for selective dissemination of scripts to specific diagnostic devices rather than broadcasting all script updates to all devices.

[0060]

In pull based embodiments, each diagnostic device would be responsible for pulling scripts from the centralized script authority. If the embodiment is purely pull based, transmission of data is independent of prompting from the centralized authority. Some pull based embodiments may include a diagnostic processor controlling and supporting one or more diagnostic devices. The centralized script authority may support a processor with a server (e.g. 156 in FIGs. 1–3 or as separate component (not shown)) for receiving requests from diagnostic devices and transmitting scripts in response to such requests. Such servers may be Web servers, FTP servers, GOPHER servers, WAIS servers, email–based file retrieval server, or other suitable server software for transmitting script files in response to a request. In other embodiments, the centralized authority may store scripts to a data store directly accessible to the diagnostic devices. A diagnostic device may pull scripts in various ways. In some embodiments, polling of the centralized authority may occur at periodic

intervals. In addition, or alternatively, polling may result from some event local to the diagnostic device such as invocation of a local software product that interfaces with the testing device and/or connection of an item for testing. In one particular embodiment, central authority may store scripts locally in a data store accessible to their own Web servers and exposed programmatically (rather than visually) for periodic collection using a Web Service approach such as Microsoft's forthcoming .NET technology. Typically, diagnostic devices will request only those scripts that apply to themselves; however, in some embodiments, all scripts may be requested and received at which point the receiving diagnostic device may sort through the received scripts for those that are applicable.

[0061]

In a particular pull based embodiment, software resides locally at each diagnostic processor. The software may act as a specialized Web client that transmits a request either periodically and/or upon occurrence of specific events (e.g. opening the local software, connection of a test item to the testing element of a diagnostic device associated with the diagnostic processor, user selecting an update script option in the software, etc.) an HTTP or HTTPS GET method request to a Web server run by a centralized script authority. The GET method request may in certain embodiments include parameters and/or cookie data designating information about diagnostic devices associated with the particular diagnostic processor; in other embodiments, the identity of the diagnostic processor originating the request may serve as a key for the Web server to access information locally about diagnostic devices associated with the particular diagnostic processor. The information may provide the server with the ability to select which scripts to transmit in response to the request. The local software may use the received scripts as a basis to generate an interface for presentation on a diagnostic display device associated with a particular diagnostic device that allows a technician to drive the particular diagnostic device to perform the scripted tests on a connected test item; in many case, the test item type and/or malfunction symptoms may also be used by the local software in the interface generation. The test item type may include generic type information as well as specific manufacturer and/or model information. In some instances, the interface may be generated independently of the specific manufacturer of the particular diagnostic

device.

[0062]   In another potential embodiment, the centralized script authority may be responsible for on-demand interface generation. In one such embodiment, local software on the diagnostic processor acts as either a specialized Web client or as a plug-in into standard browser software. The local software supports communication with the centralized script authority, the testing device, the diagnostic display, and a diagnostic data store when present. Upon receiving item type information and/or malfunction symptom information from the testing device, user data entry or both, the local software communicates information relevant to interface generation to the centralized authority using a Web server and HTTP to support the communication. The Web server at the centralized authority uses an application server to generate either the standardized interface, or a script enabling the local software to generate the standardized interface, based upon the communicated information relevant to interface generation. In some embodiments, the aggregate test data may be used to modify pre-existing static scripts based upon statistical analysis of data with respect tot he communicated information. The Web server can then return the interface as a standard HTML document. Alternatively, the Web server could transmit an XML document containing information needed by the local software to generate the final interface.

[0063]   Some embodiments supporting script management may also provide script ·generation/modification capabilities. Typically, scripts will be received from manufacturers of devices and/or authored based upon preconceived expectations of malfunctions that may occur. The aggregated diagnostic data along with such original scripts may be used to modify existing scripts or generate new scripts that take into account the realities statistically represented by the aggregated data. Such new/modified scripts may be disseminated to diagnostic devices, and potentially stored for future use/revision, by a centralized script authority as described above. The generation/modification may be provided by the script and/or system processors described above with respect to dissemination of scripts. In a push embodiment, generation of a new/modified script may serve as the trigger for dissemination of the generated script to all relevant diagnostic devices.

[0064]   In some such embodiments, script generation/modification is performed to optimize test performance for items of a given manufacturer and/or manufacturer model. Aggregated diagnostic data is analyzed to determine the most frequent malfunction according to item manufacturer/model. A new script is generated for testing an item of the particular item manufacturer/model wherein the tests for the most frequently occurring malfunction occur earlier in the scripted testing process. Where a script exists for a particular item manufacturer/model, the existing script may be modified rather than a new script being created. The new or modified script would then be available for dissemination to relevant diagnostic devices according the push or pull models described above.

[0065]   Each analyzer manufacturer may use a language to specify scripts. In the field of mobile telephone handset testing SCPI is one such language; several manufacturers have created particular dialects for use with their respective diagnostic devices. Reference to SCPI, or handset test languages generally, is purely exemplary; the principles will apply equally well for test languages used with diagnostic devices designed for use with other test item types. The administrator at the centralized script authority can manually create a script by stringing together SCPI commands to produce a test plan that tests a handset for specific functionality. Each SCPI command produces a result. Once the script is generated, the administrator typically stores the scripts in a repository such as the script data store described above. New scripts can be assigned to individual diagnostic devices on an as needed basis or provided as periodic updates. Likewise, existing scripts can be modified in a manual or automated manner.

[0066]   In some embodiments, the aggregation and reporting environment according to the present invention may be used in connection with an automated warranty submission, processing, tracking and management environment such as described in provisional U.S. Patent Application Serial No. 60/241,898, filed October 20, 2000, entitled "Service Chain Management Automation Systems and Methods." Test data for individual items as well as the aggregated test data may be used in such a warranty environment.

[0067]   Such a warranty processing component could be used to capture data required for submission of a warranty claim by each manufacturer during the process of repair. The warranty system functionality ensures that all the custom information, codes, etc. for each different mfg is captured at the point of repair. This data is then aggregated for submission to original equipment manufacturers (OEMs) for payment of warranty claims. If diagnostic data were submitted along with a claim, it would help the manufacturer validate the claim. In one embodiment, manufacturers may access the data from a central location via a suitable interface such as the Web. In such an embodiment, a typical architecture might include the various functional components depicted in FIGs. 1-3 using servers in the server cluster 150, or other servers (not shown) to perform the requisite warranty submission, processing, tracking and management functionality. Access server functionality could also be provided by servers in the server cluster 150, or other servers (not shown).

[0068]   In some embodiments, the aggregation and reporting environment according to the present invention may serve as a component within an automated service chain management system such as described in provisional U.S. Patent Application Serial No. 60/241,898, filed October 20, 2000, entitled "Service Chain Management Automation Systems and Methods."

[0069]   Throughout this application, various publications may have been referenced. The disclosures of these publications in their entireties are hereby incorporated by reference into this application in order to more fully describe the state of the art to which this invention pertains.

[0070]   The embodiments described above are given as illustrative examples only. It will be readily appreciated that many deviations may be made from the specific embodiments disclosed in this specification without departing from the invention. Accordingly, the scope of the invention is to be determined by the claims below rather than being limited to the specifically described embodiments above.